



The Engine of SOC Design

Networking Applications for Xtensa Configurable Processors

Sumit Gupta

Product Marketing Manager

Linley Tech 2007

Jan 31, 2007

Advantages of using Processors in the Networking Data Plane

■ ***If* you could use programmable processors throughout the control and data plane**

- Data plane programmability could support constantly evolving protocols
- Reduce verification time – less [or none?] hardwired logic
- Flexible platforms – multi-protocol engines could support multiple markets or emerging high-value services

■ **Most designers *think* they cannot use processors as the universal building block of networking SOCs**

- Traditional processors lack sufficient I/O capability (32b/cycle load-store bottleneck)
- Traditional fixed ISA processors lack algorithm-specific instruction sets for deeply embedded networking functions.

Xtensa Processor:

The most successful NPU you've never heard of

Xtensa Configurable Processors

Configuration options

- Mul16x16
- Floating-point unit
- Barrel Shifter
- Zero overhead looping

Extend processor

- Add register files
- multi-cycle exec units
- VLIW data path
- GPIO, FIFO interfaces

**Xtensa
Processor
Generator**

Base CPU		OCD
Apps Datapaths	Cache	Timer
Extended Registers		FPU

**Application-
optimized
processor**

RTL, EDA scripts,
test benches



**Automatically
generated
Software Tools**

C/C++ compiler
Debugger,
Simulators, RTOSes

Configurability

Check box options

- Mul16x16
- Mul32x32
- Floating-point unit
- MMU
- HiFi2 Audio Engine
- Vectra LX DSP

Drop down menus

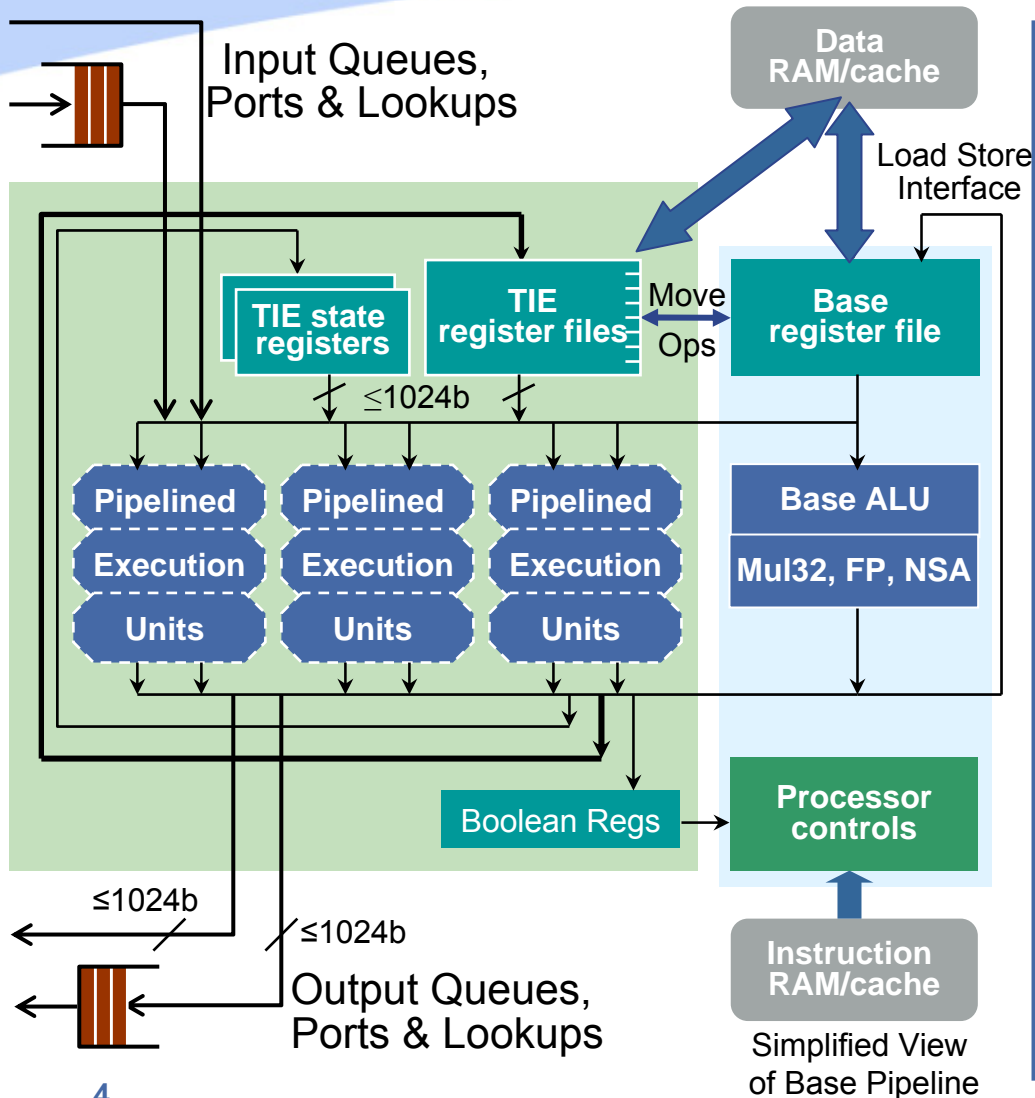
- 32 ▾ Number of Registers
- 32 ▾ Width of Local Instruction Memory Interface
- 128 ▾ Width of Local Data Memory Interface
- 32 ▾ Width of PIF System Interface
- 15 ▾ Number of Interrupts

Extensibility

Add application-specific instructions and data paths

Xtensa – Extensibility

From a RISC to a VLIW with Rich Interfaces



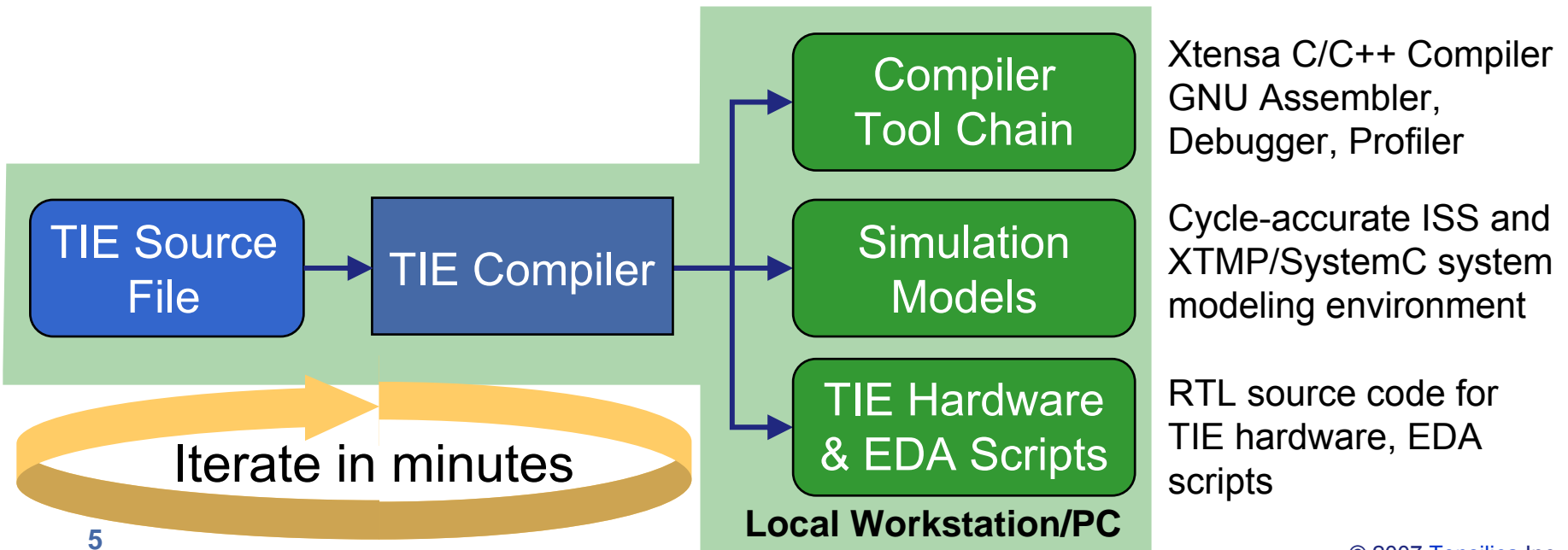
- Start with base Xtensa core
- Add functional units from menu of config options
- Add register files and state registers
 - Add corresponding new data types with automatic C/C++ compiler support
- Add up to 128-bit Load/Store instructions
- Add multi-cycle, SIMD arithmetic and logic function units
 - Up to 64 source, destination registers
- Create multi-issue VLIW datapath
- Add custom I/O ports, queues, and lookup interfaces



Generate SW Development Tools for Application-Optimized Processor in Minutes

Generate custom software tool chain in minutes

- ISS simulates new instructions
- Debugger displays designer-defined register files and state registers
- Compiler schedules custom multi-cycle TIE instructions and does register allocation of new registers





Xtensa as Conventional CPU/DSP and as RTL Alternative

Xtensa as Conventional CPU or DSP

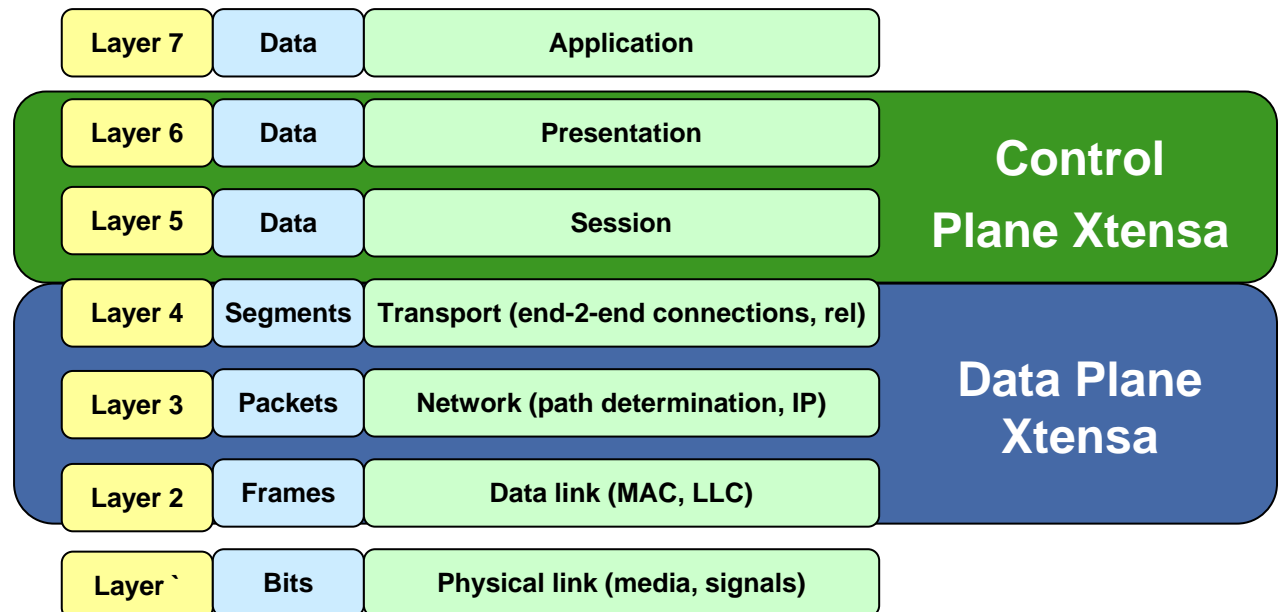
- **Build tailored processors optimized to your application**
 - Choose only the options you want
 - Not the options you are stuck with
 - Base processor configuration is only 20K gates
- **Accelerate hot spots in applications by adding application-specific instructions**
 - Not by increasing MHz (and hence power)
- **Mixed 24/16-bit ISA gives 25-50% higher code density than 32/16-bit architectures**
- **Automatically generated, pre-verified RTL**
- **Post-silicon flexibility and bug fixes**
 - Reprogram processor to adapt to upgrades and bugs in algorithms

Xtensa as RTL Alternative

- **Create optimized task engines using Xtensa -vs- hardwired RTL blocks**
- **Create a data path similar to that in a hardwired RTL block**
 - Easy to add multi-cycle, complex execution units
 - Automatic generation of pre-verified RTL
 - Verify only the input-output relationship of extensions specification
- **Reduce the verification effort and time**
 - In RTL design, >90% of the bugs & only 10% of the logic is in the control FSM
 - Use Xtensa to map the control FSM to software on the processor
- **Very low power solution**
 - Automatic RTL generation -> very fine grained clock gating
 - To achieve the same amount of clock gating in hand-optimized RTL requires high verification time and resources

Xtensa in use on OSI layer 2, 3 and 4 applications

- TOE-engines, MACs, bridging, routing
- Packet classification, AAL5 SAR, and IPSEC (3DES)
- Network speeds from DS-1 to 96 Gb/s



Xtensa Data Throughput: Orders of Magnitude Higher –vs– Conventional CPUs

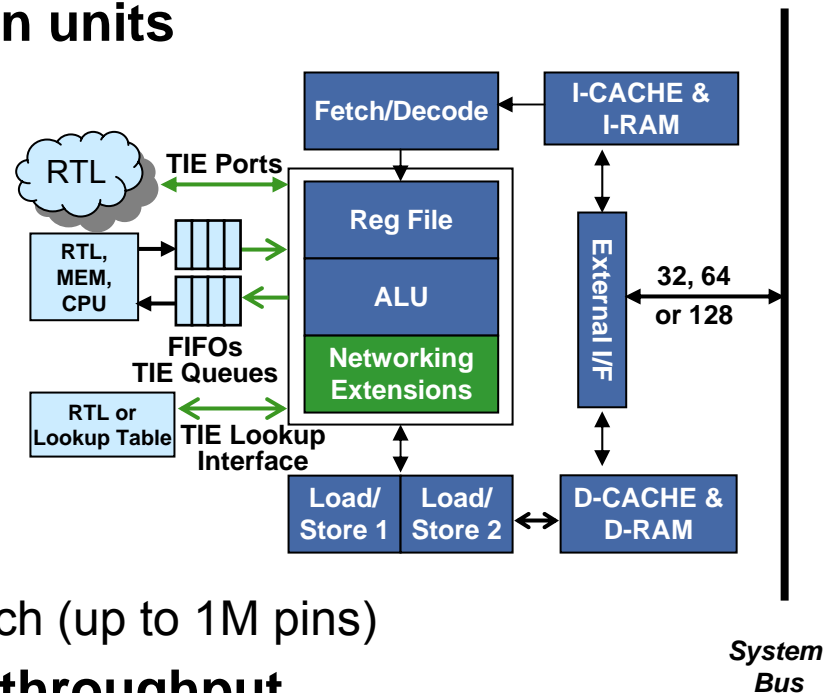
- ▀ **Networking thrives on speeds and feeds**
 - Traditional processors are load/store bound (32b per cycle)
- ▀ **Conventional CPUs are limited by System bus bandwidth**
- ▀ **Xtensa processors allow designer-specified data interfaces wired directly into computation units**

- Ports (like GPIO)
- Queues (FIFO interfaces)
- Lookups (Address out, data in)

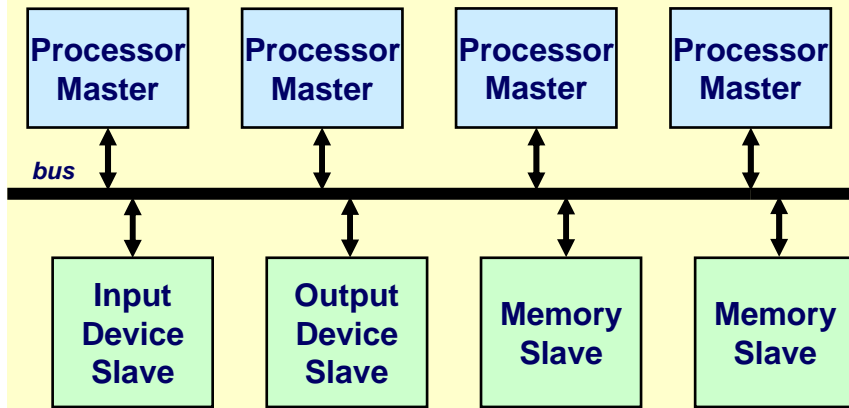
Interface characteristics

- Designer-defined instructions process interface data
- All interfaces can be active in a single Xtensa instruction
- Maximum of 1024 interfaces w/ maximum width of 1024 bits each (up to 1M pins)

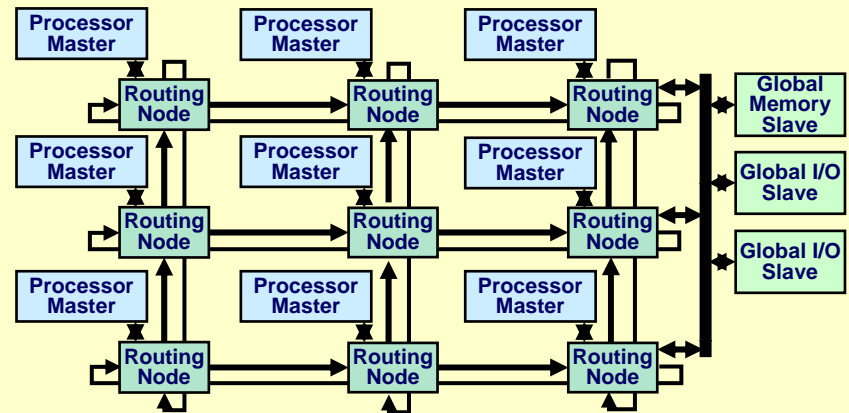
▀ **@500MHz, that's a lot of data throughput**



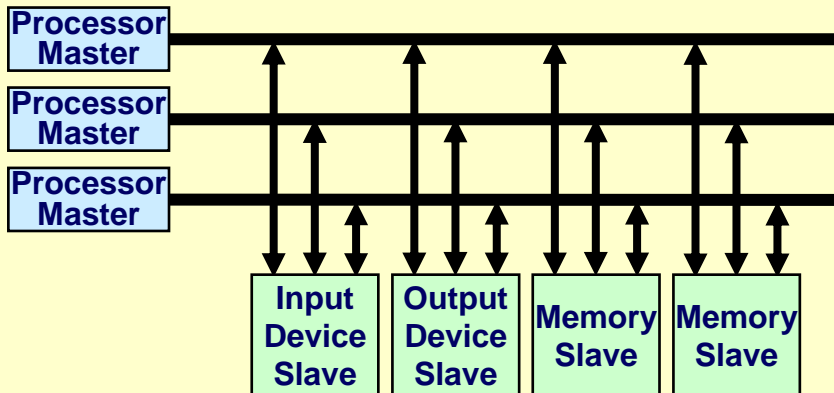
Shared Bus



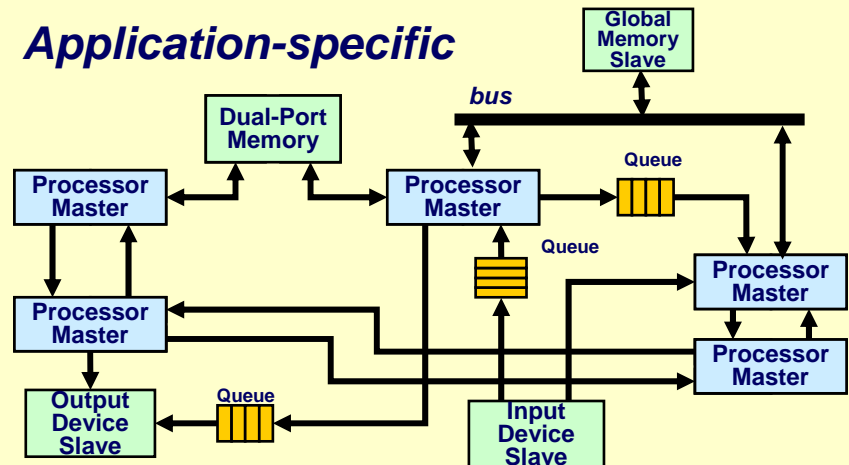
On-chip Routing Network



Cross-Bar



Application-specific



Control Plane Example



Tensilica Diamond 570T

An Instance of a High-Performance Xtensa

Half the Power and Area of ARM11

- 3-issue VLIW (FLIX) CPU - 16/24/64-bit instructions with modeless switching
- Ultra-high performance in small footprint: 0.59 mm² cell area, 525 MHz (in 90nm GT)
- 32-bit input Queue and 32-bit output Queue
 - Ultra-high bandwidth direct access directly from Pipeline (no load/stores required)

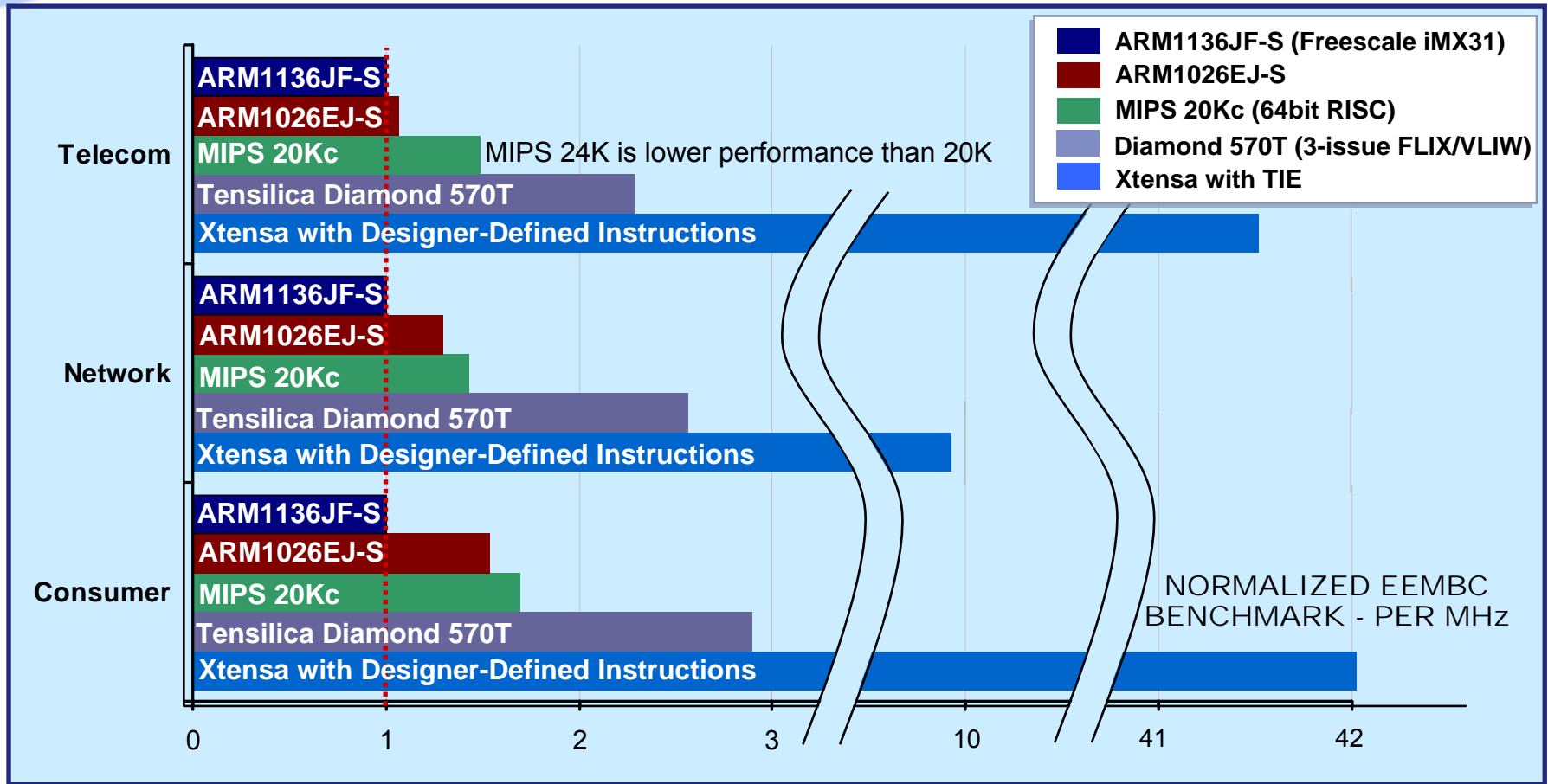
	ARM 1156T2-S	Tensilica Diamond 570T	ARM 1136J-S
Max Freq. (0.13u G) worst case	335 MHz	250 MHz	330 MHz
Area (post-layout)	2.40 mm ²	1.58 mm ²	2.85 mm ²
mW / MHz (0.13u G) typical conditions	0.45	0.17	0.60
# Pipeline Stages	9	5	8
Instruction Width	16/32 bit	16/24/64 bit 3-issue	16/32 bit
High Throughput Data Queues	No	Yes 32b In / 32b Out	No
Direct Ports	No	32 In / 32 Out	No

Data on ARM products taken from ARM public website, Dec2005, for TSMC 0.13G process
All speed, power, area metrics are subject to variation based on user's design and fab choices.



High Performance Xtensa & Diamond Beat ARM & MIPS on Networking Benchmarks

EEMBC Benchmark Results



All scores are Simulations of Licensable cores.

All scores are EEMBC/ECL Certified. All scores "out of the box" except Xtensa with TIE (optimized) Per-MHz certified benchmark scores normalized to ARM = unit score of 1 for suitability in graphing.

Competitive Data as of June 2006. Source: www.eembc.org



Data Plane Examples Using Xtensa Processors

- **IPSEC 3DES**

- **Packet Classification**

In the Extra Slides

- ***AAL5 Segmentation and Reassembly***

- ***TCP Offload Engines***

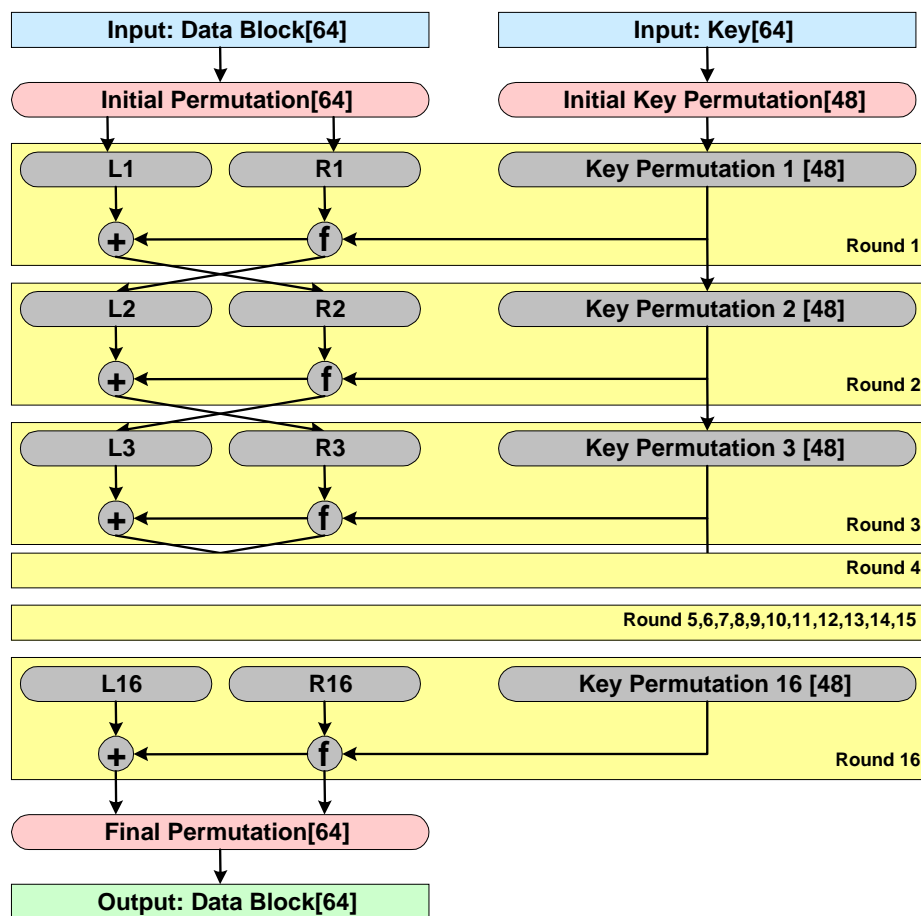
- **Traditional processors and NPUs are limited to a fixed ISA**
- **Xtensa's ISA: SOC designers invent new instructions to boost performance by 2x-100x or more**
- **Example:**
 - IPSEC triple DES one clock tick per 2-DES rounds; 24-Xtensa clock ticks per 3DES cipher-block
 - Normally 1000's of instructions per 3DES cipher-block
 - Code density increases
 - These new instructions become native to the Xtensa core
- **Tensilica's XPRES Compiler analyzes software**
 - Recommends new instructions to boost performance
 - VLIW/FLIX multi-issue instruction (super-scalar, up to 15 concurrent issues)

3DES algorithm computationally intensive at the bit level

Consider a single DES block diagram

- Each round contains
 - 2-32 bit values L,R
 - Two (+) 32-bit exclusive-or functions
 - One (f) bit scrambler
 - One key permutation unit

Single DES Block Diagram



3DES algorithm computationally intensive at the bit level

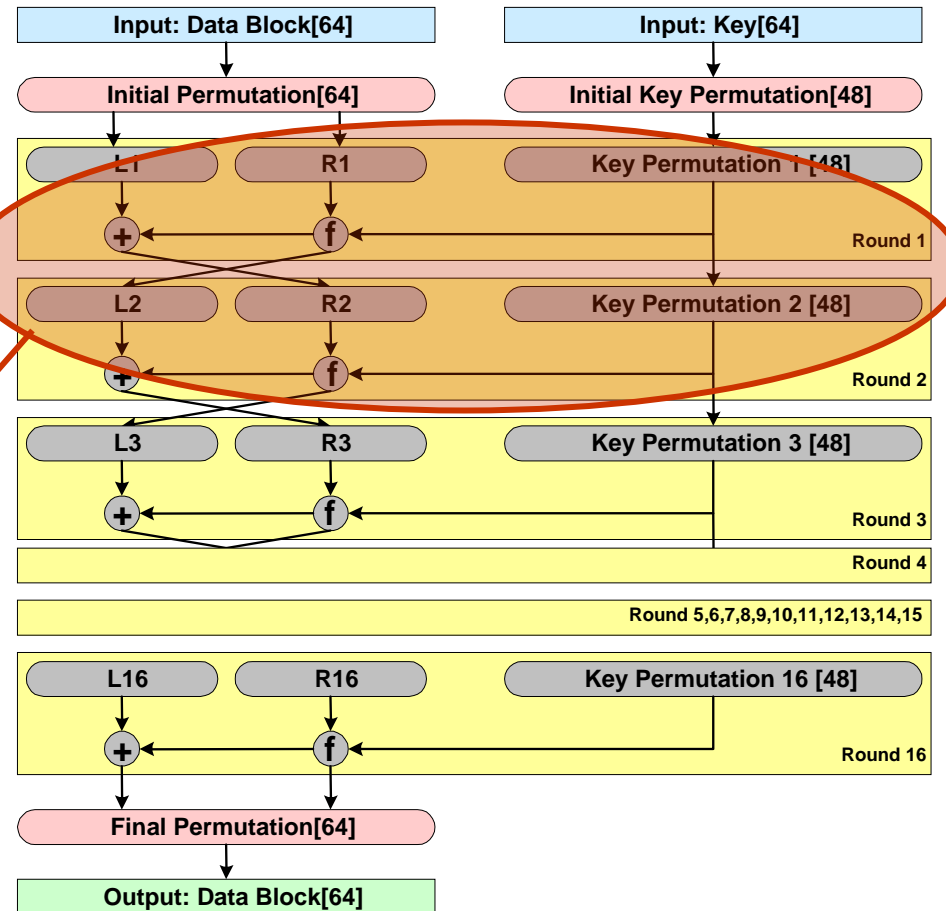
Consider a single DES block diagram

- Each round contains
 - 2-32 bit values L,R
 - Two (+) 32-bit exclusive-or functions
 - One (f) bit scrambler
 - One key permutation unit

1 TIE instruction

- Computes 2-rounds in one cycle
- Computes XOR (+), scrambler (f), and key permutation in parallel
- 64-bit processing

Single DES Block Diagram



■ Performance (250MHz TSMC 130nm LV)

- 8-clocks per DES cipher-block
 - Data rate: 31.250 million DES blocks/second (2 Gb/s data rate)
- 24-clocks per 3DES cipher-block
 - Data rate: 10.416 million 3DES blocks/second (666 Mb/s data rate)

■ 64-bit local memory interfaces

- Data remains in memory local to processor for upstream/down stream processing

■ Single Xtensa processor for encryption and decryption

- Xtensa 3DES performance rivals RTL

Example #2: Packet Classification

■ 20Gbps Ethernet line rate

- Minimum packet size of 84 bytes (64 bytes packet, 20 bytes overhead)
- Approx 29.6M packets/sec
- Use of TCam for lookup function

■ Use a multi-processor Xtensa solution

- VLIW configuration
- Instructions to accelerate packet processing
- Rich I/O interfaces to achieve high throughput

■ Xtensa processor subsystem

- 128-bit memory interfaces
- TIE Queue interfaces for communication

■ 4 TIE Queues

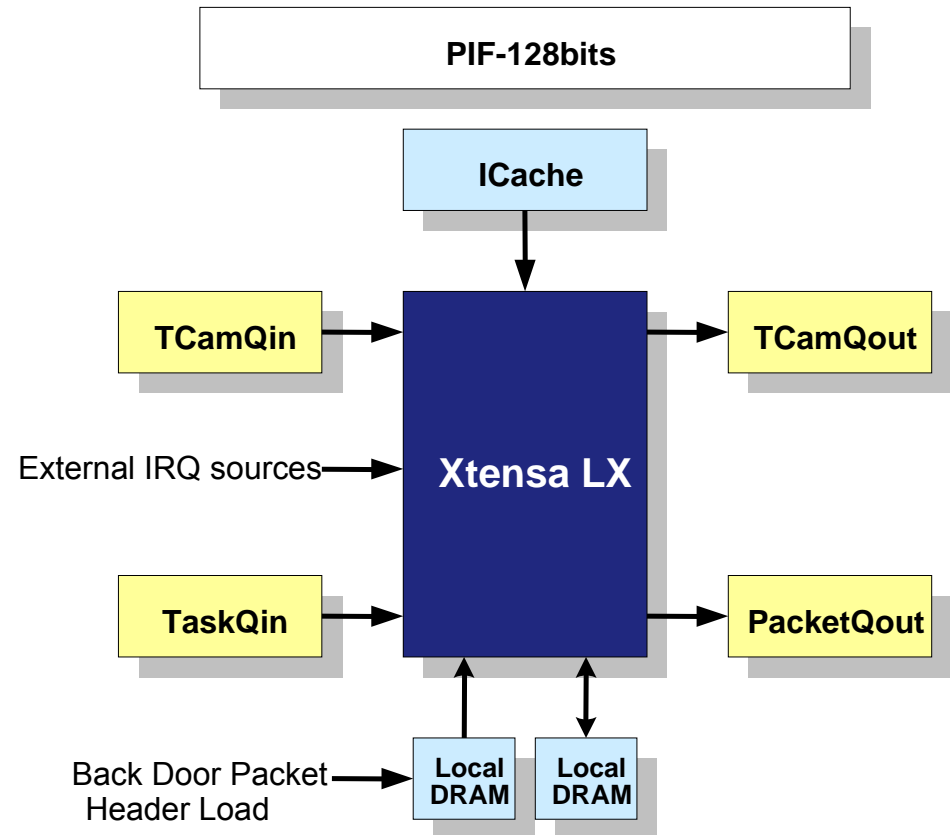
- TCamQout, TCamQin
- TaskQin, PacketQout

■ 2 local data rams

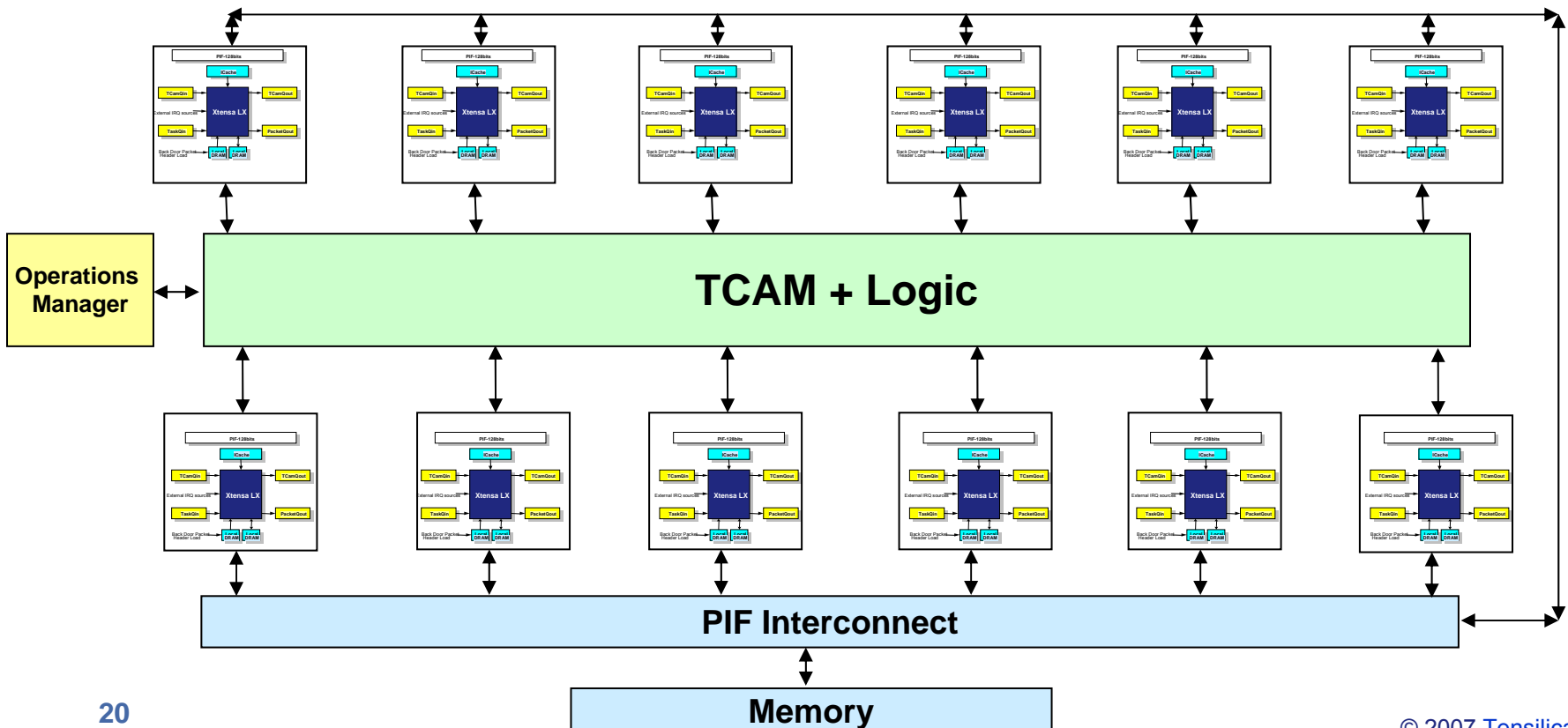
- program vars
- packet headers

■ TIE Enhanced ISA

- FLIX (VLIW) instructions
- Parse packet header to determine packet type
- Classify packet based on header field criteria
- Rewrite header to downstream engine



- To achieve 20 Gbps rate, 12 processor-subsystems are required
 - 21 mm² in 90nm technology; 400 MHz



Tensilica's Networking & Communications Customers

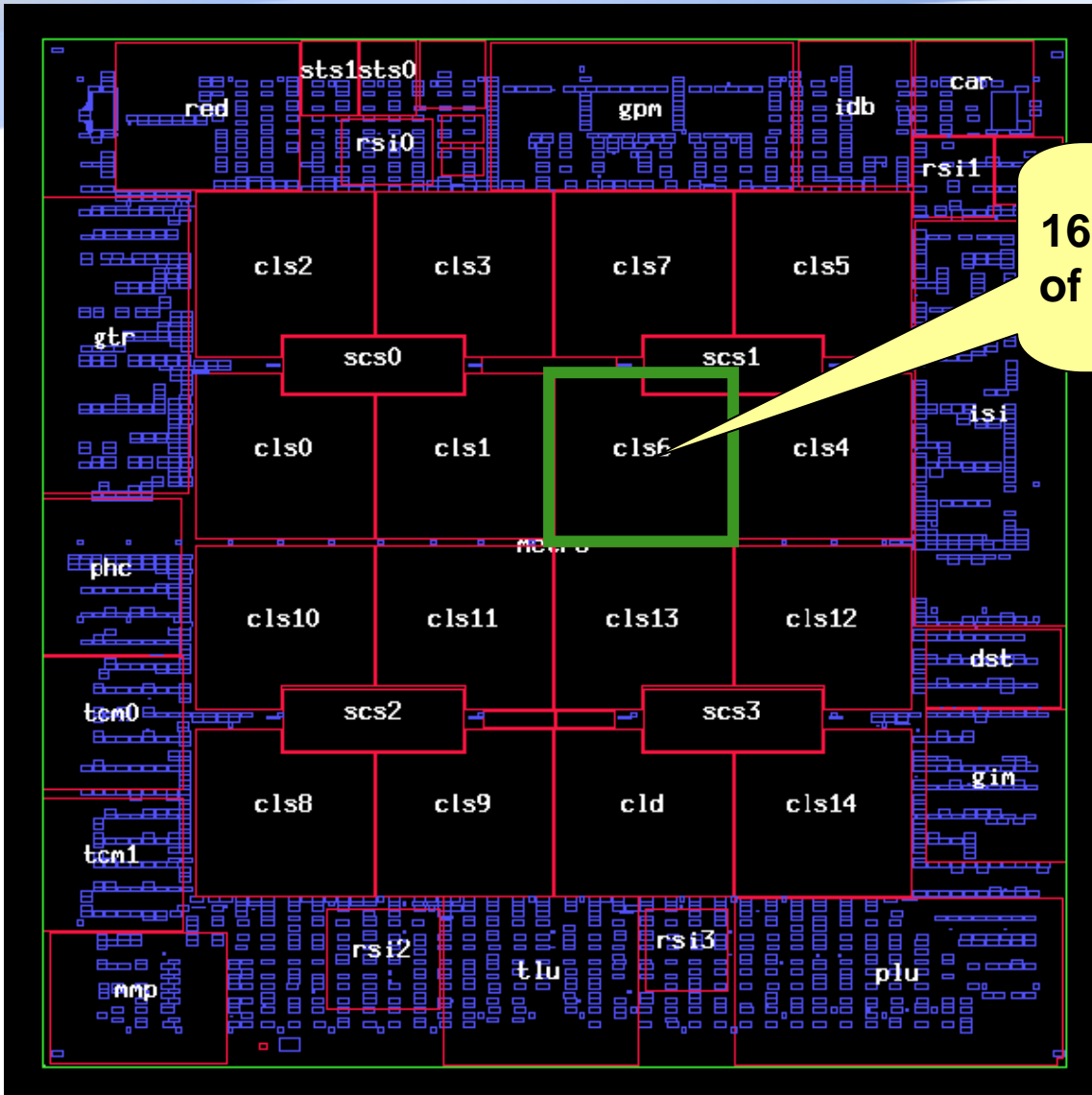
Networking and Communications






Other Industries



SPP Chip of Cisco's CRS-1



**16 PPE Clusters
of 12 PPEs each**

-  **188 Xtensa cores on each SPP chip**
-  **2 SPP chips per linecard**
-  **96 Gbps BW packets in & out**

From Will Eatherton's ANCS '05 presentation



In Summary

- **Tensilica technology proven for networking and communication applications**
 - Products shipping in volume

- **Xtensa technology addresses NPU needs:**
 - Programmability
 - Line speed data throughput through GPIO and FIFO interfaces
 - High performance data processing

- **Highly efficient: performance and energy**

- **Very easy to use with complete toolset**



Extras

Cisco's CSR-1 92-Terabit router uses 188 Xtensa cores per SOC

Bill Jennings, vice president of engineering in the routing technology group at Cisco Systems, a Tensilica customer and investor, says the Xtensa core is superior to others Cisco has tested, in terms of several technical factors, including average instruction size and memory performance. His team is just one of several using Xtensa at Cisco. If Tensilica didn't exist, Jennings says, Cisco would have to develop the necessary high-performance processors itself. "Even though we have the hardware expertise, we don't have the software and tool set Tensilica does."

Electronic Business magazine; Nov. 1, 2003



"The Silicon Packet Processor (SPP) represents the heart of the CRS-1's forwarding engine. The SPP is a software programmable network processor with a massively parallel array of 188 embedded CPU cores."

Cisco website, June 2004

Ikanos: VDSL Modem Engine



Sept 22, 2005 News
Ikanos completes successful IPO.

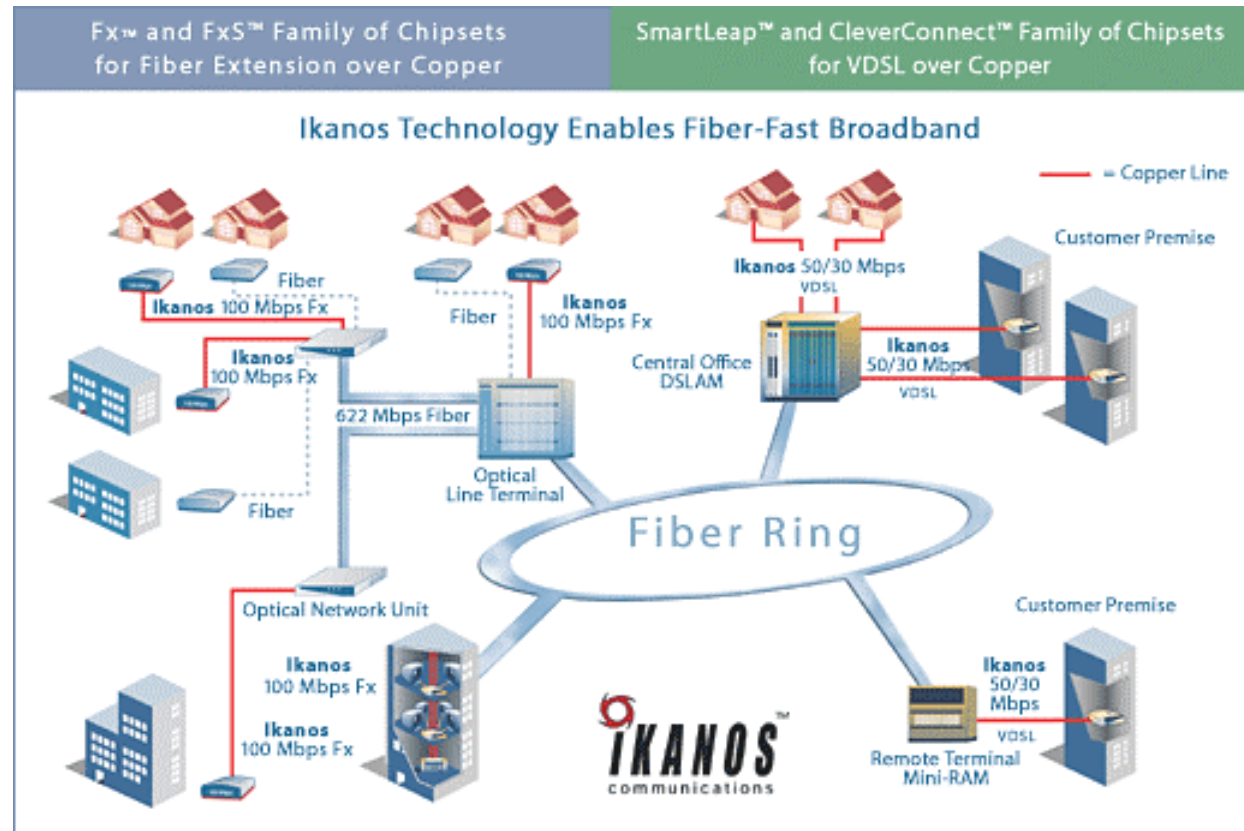


“We had to ensure that Ikanos’ solution was very flexible and future proof.” We met our performance targets and significantly reduced the development time for this chip”.

Anoop Khurana
 VP Engineering, Ikanos

Fx™ and FxS™ Family of Chipsets
 for Fiber Extension over Copper

SmartLeap™ and CleverConnect™ Family of Chipsets
 for VDSL over Copper



NEC iStorage NV8200 Series

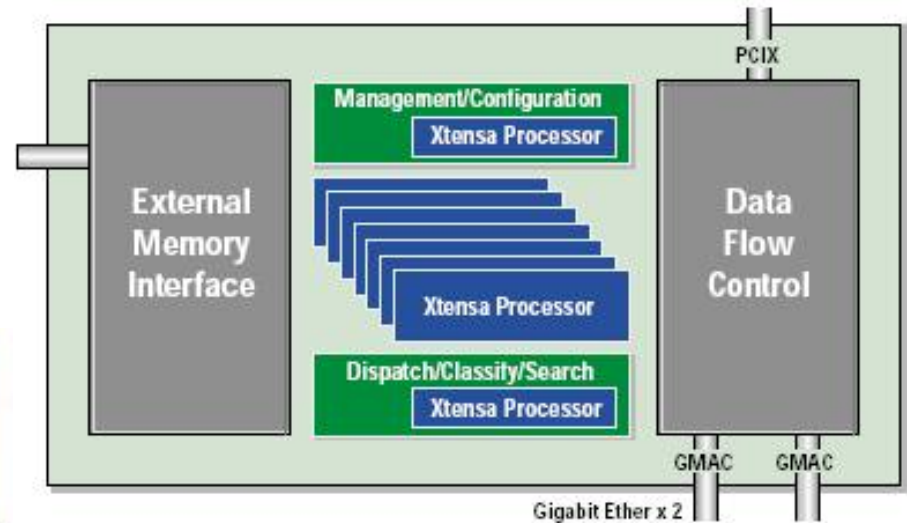
TCP/IP offload engine (TOE) chip w/ 10 Xtensa processors
One TOE engine per blade



NV8220

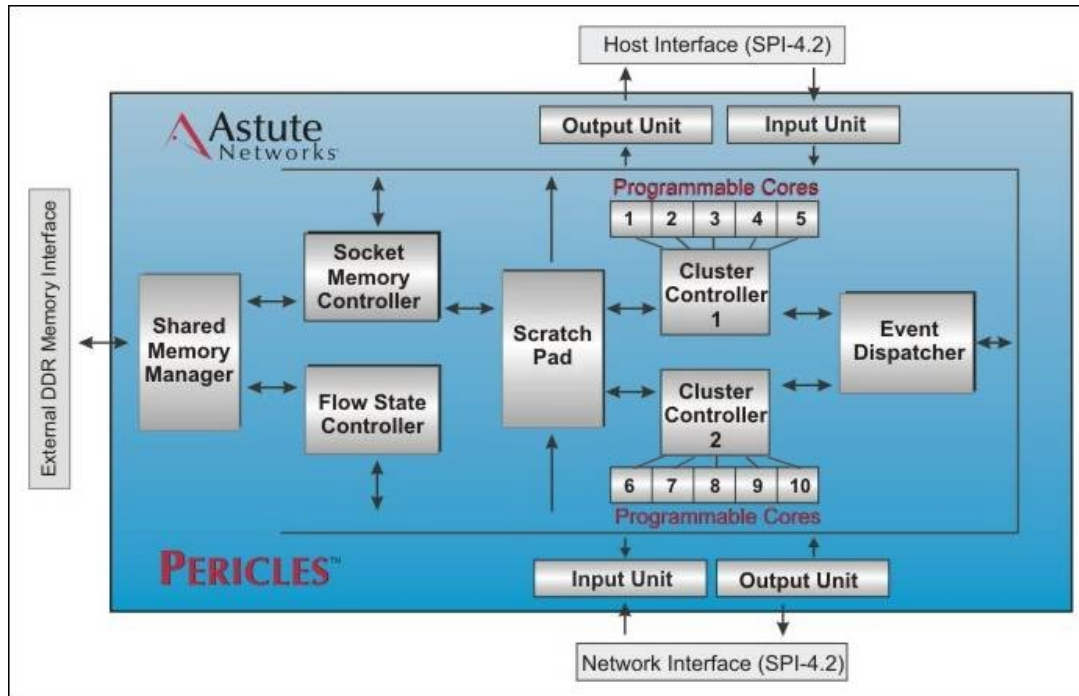


NV8210



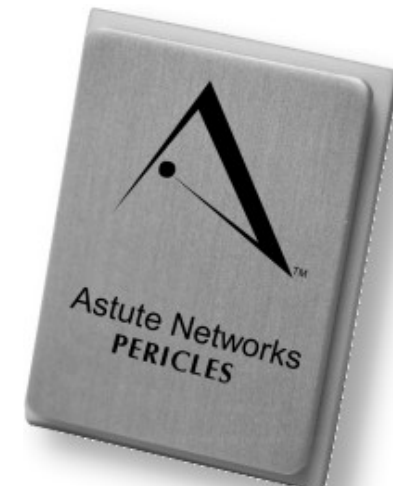
Astute Networks Pericles Network Storage Processor

- 10 Xtensa processors running @ 267 MHz
- 10 Gbps full duplex performance
- Multiple storage protocol support - TCP, iSCSI, Fibre Channel, SCSI, FCIP or proprietary formats - simultaneously



 **Astute Networks™**

PERICLES™





NetScreen-ISG 2000

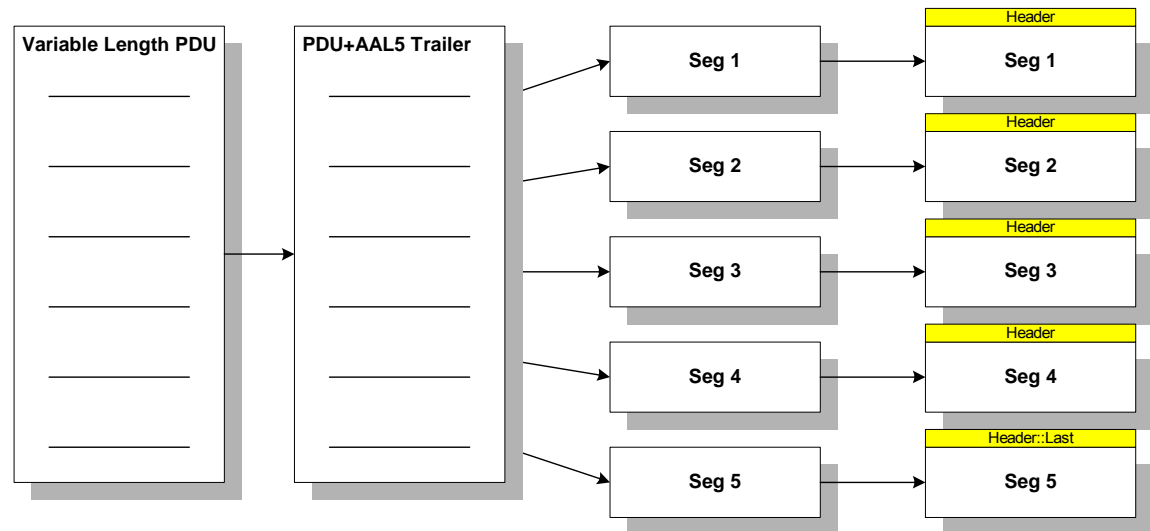
- ▀ **NetScreen-ISG 2000 Security Gateway**
- ▀ **GigaScreen3 security ASIC with 2 Xtensa processors**

■ AAL5 Segmentation and Reassembly

■ Packets (PDUs) to ATM cells

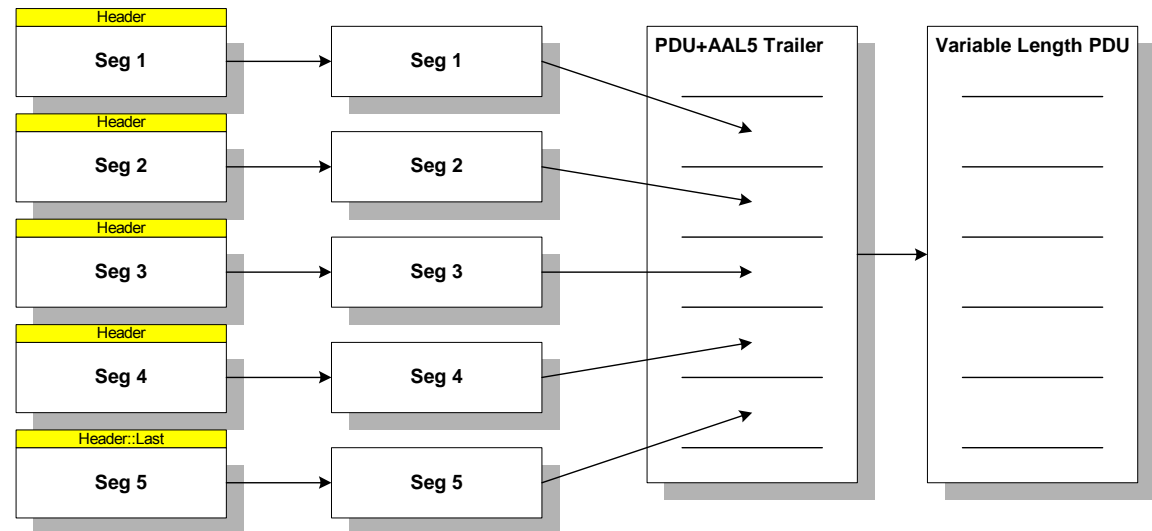
- Calculates trailer information
- Pre-pend and ATM header and forwards to output queue

■ 128-bit Memory Interface



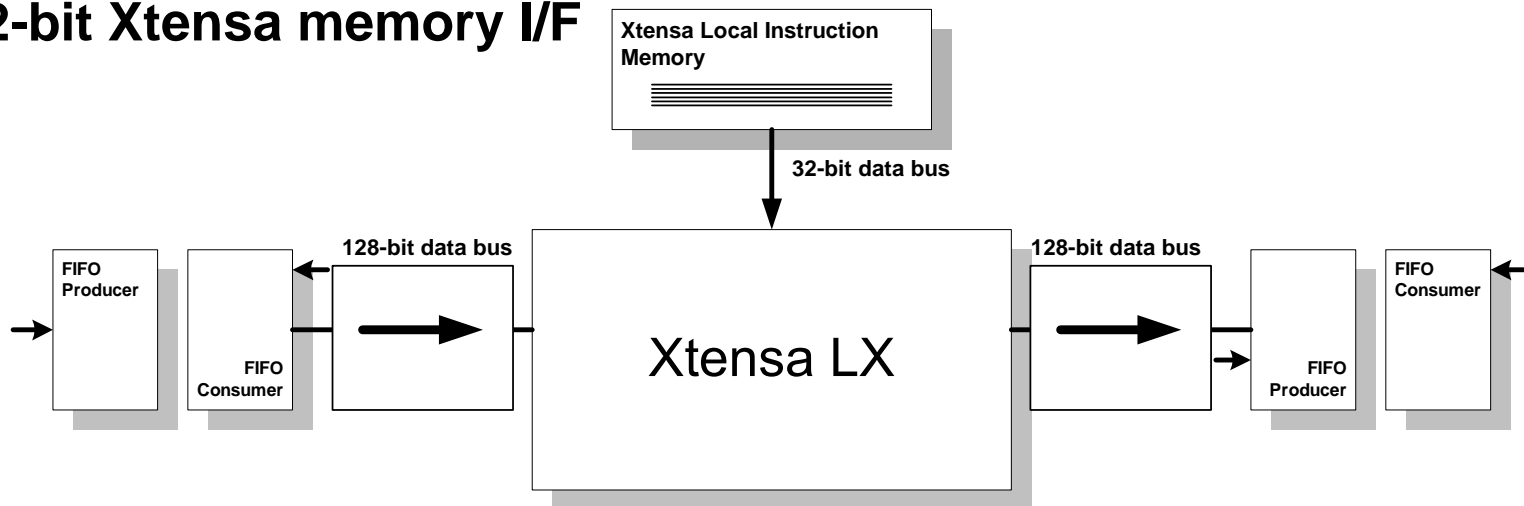
ATM cells to Packets (PDU)

- Strip off the ATM header
- Aggregate ATM payloads
- Validate AAL5 Trailer and strip

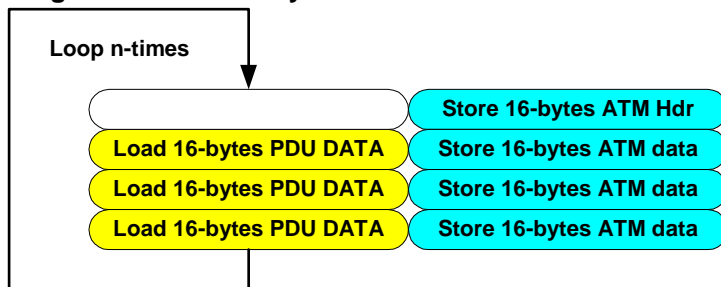


Xtensa LX AAL5 I/O System

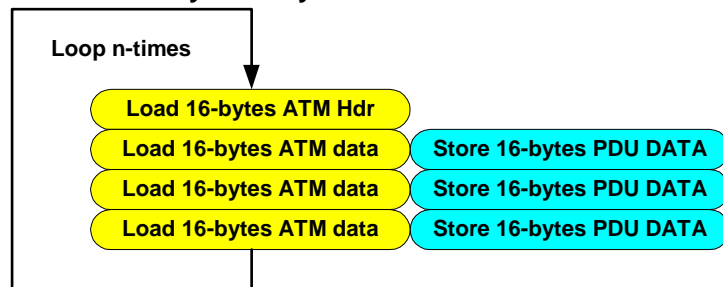
- 1 Input Queue, 1 Output Queue
- @250Mhz max data throughput 32Gbs (Queue)
- 32-bit Xtensa memory I/F



Segmentation Memory Access



Assembly Memory Access





LX AAL5 Segmentation Algorithm

```
void AAL5_Segm_TIE_q()
{
    u32b i, loopCnt;

    xt_AAL5_init();
    loopCnt = xt_AAL5_getHdrTpl();

//ATM cell generation
    for (i=0;(i < loopCnt);i++)
    {
        xt_AAL5_putAtmHdr(0);           //not last cell
        xt_AAL5_putPDU(0);             //CRC pdu and send
        xt_AAL5_putPDU(1);             //CRC pdu and send
        xt_AAL5_putPDU(1);             //CRC pdu and send
    }
//LAST ATM cell generation
    xt_AAL5_putAtmHdr(1);             //Last ATM Cell
    xt_AAL5_putPDU(0);               //CRC pdu and send
    xt_AAL5_putPDU(1);               //CRC pdu and send
    xt_AAL5_putPDU(3);               //CRC pdu stuff Trailer and send...
}
```



LX AAL5 Re-assembly Algorithm

```
u32b AAL5_Assm_TIE_q()  
{  
    u32b status;  
    u32b i;  
  
    status = xt_AAL5_init();  
    while (status == 0)  
    {  
        xt_AAL5_getHdrTpl();  
        xt_AAL5_putPDU(0);  
        xt_AAL5_putPDU(1);  
        status = xt_AAL5_putPDU(2);  
    }  
    return(status);  
}
```

Performance (TSMC 130nm LV)

Packet Size	Segment Cycles	bits/second @133MHz	bits/second @300MHz
40	17	2.50Gbs	5.65Gbs
1500	236	6.76Gbs	15.25Gbs
9600	1422	7.18Gbs	16.20Gbs

Packet Size	Assembly Cycles	bits/second @133MHz	bits/second @300MHz
40	17	2.50Gbs	5.65Gbs
1500	340	4.70Gbs	10.59Gbs
9600	2020	5.06Gbs	11.41Gbs

Bypassing the Choke Point (System bus)

Xtensa LX2 Port, Queue, & Lookup Interfaces

- **Connect wires, queues, RAM tables directly to processor data path**
 - Use an Xtensa processor instead of hardwired RTL without changing the interface to the rest of the RTL blocks
 - All 100% automatically generated, modeled and pre-verified
- **Read/write to I/O ports, queues, lookups & operate on data in same cycle**
 - No load or store instructions required to directly access data
- **Synchronization is built-in**
 - Automatic processor stall on empty input queue or full output queue
- **Up to 1M designer-defined I/O**
 - 350 Terabit/sec throughput (1024 ports @ 1024 pins @ 350 MHz)

